CLAIMS

What is claimed is:

1. A method of verifying the trustworthiness of a browser, comprising:

transmitting an electronic document requiring signature from a first user computer to a second user computer;

electronically signing the electronic document at the second user computer to create a first digital signature;

including as an attribute of the first digital signature a second digital signature, the second digital signature verifying the authenticity of one or more components running in an environment of the browser on the second user computer;

transmitting the signed electronic document from the second user computer to the first user computer;

authenticating the second digital signature.

2. The method of claim 1, further comprising determining whether the entity that executed the second digital signature is authorized to certify the trustworthiness of the one or more components.

3. The method of claim 1, wherein the attribute is a signed attribute.

4. The method of claim 1, wherein the attribute is an authenticated attribute.

5. The method of claim 1, wherein the authenticating comprises verifying the authenticity of the second digital signature.

6. The method of claim 5, wherein the authenticity of the second digital signature is verified using a digital certificate.

7. The method of claim 1, wherein the authenticating comprises comparing a hash of the one or more components running in the browser environment included in the second digital signature to a known-good hash of the one or more components running in the browser environment.

8. The method of claim 1, wherein the authenticating is performed by the first user computer.

9. The method of claim 1, wherein the authenticating is performed by a computer maintained by a participant.

10. The method of claim 1, wherein the authenticating is performed by an independent entity that is not a participant.

11. The method of claim 1, wherein the authenticating is performed by the second user computer.

12. The method of claim 1, wherein an unsigned component running in the browser environment of the second user computer is included as an attribute of the first digital signature.

13. The method of claim 12, wherein the unsigned component is copied from RAM of the second user computer.

14. The method of claim 12, wherein the unsigned component is copied from non-volatile memory of the second user computer.

15. The method of claim 1, wherein a hash of one or more signed browser components running on the second user computer is included as an attribute of the first digital signature.

16. The method of claim 15, wherein the one or more signed components are copied from RAM of the second user computer.

17. The method of claim 15, wherein the one or more signed components are copied from non-volatile memory of the second user computer.

18. A method of verifying the trustworthiness of a browser comprising:

creating a first set of hashes, the first set of hashes comprising a hash of the browser at a first point in time, the first set of hashes being a known-good set of hashes;

determining the status of the browser by:

creating a second set of hashes, the second set of hashes comprising a hash of the browser at a second point in time;

verifying the second set of hashes to ensure that each hash was created by a trusted source; and

comparing the first set of hashes to the second set of hashes.

19. The method of claim 18, wherein the step of determining is performed at a second, subsequent point in time.

20. The method of claim 18, wherein the step of determining further comprises verifying the status of the browser if the first set of hashes matches the second set of hashes.

21. The method of claim 18, wherein the step of determining further comprises determining that the status of the browser is bad if the first set of hashes does not match the second set of hashes.

22. The method of claim 18, wherein the step of determining further comprises determining that the status of the browser is unknown if it can not be determined that a hash in the second set of hashes was created by a trusted source.

23. The method of claim 18, wherein the step of determining further comprises determining that the status of the browser is unknown if it is determined that a hash in the second set of hashes was not created by a trusted source.

24. The method of claim 18, wherein the first set of hashes is maintained by a trusted entity, and further comprising the steps of:

receiving from a requestor a request to determine the trustworthiness of the browser, the request including the second set of hashes;

generating a report about the status of the browser based on a result of the determining step; and

transmitting the report to the requestor.

25. The method of claim 24, wherein the steps of receiving, determining, generating, and transmitting are performed by the trusted entity.

26. The method of claim 18, wherein the second set of hashes comprises one or more hashes of browser components at a second point in time.

27. The method of claim 26, wherein the first set of hashes comprises hashes at a first point in time corresponding to the hashes in the second set of hashes.

28. The method of claim 27, wherein the step of determining is performed at a second, subsequent point in time.

29. The method of claim 27, wherein one or more of the hashes in the second set of hashes has been signed by a trusted source.

30. The method of claim 29, wherein the step of verifying further comprises for verifying that a hash in the second set of hashes was created by a trusted source by verifying the signature on the hash.

31. The method of claim 18, wherein the browser status request is received from a first customer seeking to verify the trustworthiness of a browser running on a computer in the possession of a second customer.

32. The method of claim 31, wherein in the first customer and the second customer are parties to a transaction.

33. The method of claim 32, wherein the first customer is a buyer and the second customer is a seller in the transaction.

34. The method of claim 31, wherein the second customer disaffirms the transaction based on the status of the browser.

35. A system for providing trusted browser verification comprising:

a trusted verifier;

means for maintaining by the trusted verifier a first set of hashes, the first set of hashes comprising a hash of a browser, the first set of hashes being a known-good set of hashes;

means for receiving by the trusted verifier a browser status request, the browser status request including a second set of hashes, the second set of hashes comprising a second hash of the browser;

means for verifying by the trusted verifier that each hash in the second set of hashes was created by a trusted source; and

means for determining by the trusted verifier the status of the browser based on the first set of hashes and the second set of hashes.

36. The system of claim 35, wherein the trusted verifier determines the status of the browser by comparing the first set of hashes with the second set of hashes.

37. The system of claim 36, wherein the trusted verifier verifies the status of the browser if the first set of hashes matches the second set of hashes.

38. The system of claim 36, wherein the means for determining determines that the status of the browser is bad if the first set of hashes does not match the second set of hashes.

39. The system of claim 36, wherein the means of determining determines that the status of the browser is unknown if it can not be determined that a hash in the second set of hashes was created by a trusted source.

40. The system of claim 36, wherein the means of determining determines that the status of the browser is unknown if it is determined that a hash in the second set of hashes was not created by a trusted source.

41. The system of claim 35, wherein the second set of hashes comprises one or more hashes of browser components at a second point in time.

42. The system of claim 41, wherein the first set of hashes comprises hashes at a first point in time corresponding to the hashes in the second set of hashes.

43. The system of claim 42, wherein the step of determining is performed at a second, subsequent point in time.

44. The system of claim 42, wherein one or more of the hashes in the second set of hashes has been signed by a trusted source.

45. The system of claim 44, wherein the means for verifying verifies that a hash in the second set of hashes was created by a trusted source by verifying the signature on the hash.

46. The system of claim 35, wherein the browser status request is received from a first customer seeking to verify the trustworthiness of a browser running on a computer in the possession of a second customer.

47. The system of claim 46, wherein in the first customer and the second customer are parties to a transaction.

48. The system of claim 47, wherein the first customer is a buyer and the second customer is a seller in the transaction.

49. The method of claim 46, wherein the second customer disaffirms the transaction based on the status of the browser.

50. In a system comprising a root entity, a first participant, a second participant, a first customer of the first participant, a second customer of the second participant, a method for verifying the trustworthiness of a browser in possession of the first customer comprising:
a) maintaining at a trusted verifier a first set of hashes, the first set of hashes comprising a first hash of the first customer's browser;
b) generating by the first customer a second set of hashes, the second set of hashes comprising a second hash of the first customer's browser;
c) transmitting by the first customer the second set of hashes to the second customer;
d) generating by the second customer a browser status request, the browser status request including the second set of hashes;
e) transmitting by the second customer the browser status request to the second participant;
f) forwarding by the second participant the browser status request to the trusted verifier;
g) determining by the trusted verifier a status of the first customer's browser;
h) generating by the trusted verifier a browser status response;
i) forwarding by the trusted verifier the browser status response to the second participant; and

j) transmitting by the second participant the browser status response to the second customer.

51. The method of claim 50, wherein the trusted verifier determines the status of the browser by comparing the first set of hashes with the second set of hashes.

52. The method of claim 51, wherein the status of the browser is verified if the first set of hashes matches the second set of hashes.

53. The method of claim 51, wherein the status of the browser is one of good, bad, or unknown.

54. The method of claim 50, wherein the trusted verifier verifies that each hash in the second set of hashes was created by a trusted source.

55. The method of claim 54, wherein the status of the browser is verified if the first set of hashes matches the second set of hashes.

56. The method of claim 54, wherein the status of the browser is one of good, bad, or unknown.

57. The method of claim 50, wherein in the first customer and the second customer are parties to a transaction.

58. The method of claim 57, wherein the first customer is a buyer and the second customer is a seller in the transaction.

59. The method of claim 58, wherein the second customer disaffirms the transaction based on the status of the browser.

60. The method of claim 50, wherein the root entity establishes a set of operating rules for the system.

61. The method of claim 50, wherein the first participant is a financial institution.

62. The method of claim 50, wherein the second participant is a financial institution.

63. The method of claim 50, wherein the first participant comprises a transaction coordinator for processing browser status requests.

64. The method of claim 50, wherein the second participant comprises a transaction coordinator for processing browser status requests.

65. The method of claim 50, wherein the trusted verifier is an integrated component of the first participant.

66. The method of claim 50, wherein the trusted verifier is an integrated component of the second participant.

67. The method of claim 50, wherein the trusted verifier is a distinct entity from the first and second participants.

68. A system for verifying the trustworthiness of a browser in possession of a first customer comprising:

a root entity;

a first participant;

a second participant;

the first customer of the first participant;

a second customer of the second participant;

means for maintaining at a trusted verifier a first set of hashes, the first set of hashes comprising a first hash of the first customer's browser;

means for generating by the first customer a second set of hashes, the second set of hashes comprising a second hash of the first customer's browser;

means for transmitting by the first customer the second set of hashes to the second customer;

means for generating by the second customer a browser status request, the browser status request including the second set of hashes;

means for transmitting by the second customer the browser status request to the second participant;

means for forwarding by the second participant the browser status request to the trusted verifier;

means for determining by the trusted verifier a status of the first customer's browser;

means for generating by the trusted verifier a browser status response;

means for forwarding by the trusted verifier the browser status response to the second participant; and

means for transmitting by the second participant the browser status response to the second customer.

69. The system of claim 68, wherein the trusted verifier determines the status of the browser by comparing the first set of hashes with the second set of hashes.

70. The system of claim 69, wherein the status of the browser is verified if the first set of hashes matches the second set of hashes.

71. The system of claim 69, wherein the status of the browser is one of good, bad, or unknown.

72. The system of claim 68, wherein the trusted verifier verifies that each hash in the second set of hashes was created by a trusted source.

73. The system of claim 72, wherein the trusted verifier determines the status of the browser by comparing the first set of hashes with the second set of hashes.

74. The system of claim 73, wherein the status of the browser is verified if the first set of hashes matches the second set of hashes.

75. The system of claim 74, wherein the status of the browser is one of good, bad, or unknown.

76. The system of claim 68, wherein in the first customer and the second customer are parties to a transaction.

77. The system of claim 68, wherein the first customer is a buyer and the second customer is a seller in the transaction.

78. The system of claim 68, wherein the second customer disaffirms the transaction based on the status of the browser.

79. The system of claim 68, wherein the root entity establishes a set of operating rules for the system.

80. The system of claim 68, wherein the first participant is a financial institution.

81. The system of claim 68, wherein the second participant is a financial institution.

82. The system of claim 68, wherein the first participant comprises a transaction coordinator for processing browser status requests.

83. The system of claim 68, wherein the second participant comprises a transaction coordinator for processing browser status requests.

84. The system of claim 68, wherein the trusted verifier is an integrated component of the first participant.

85. The system of claim 68, wherein the trusted verifier is an integrated component of the second participant.

86. The system of claim 68, wherein the trusted verifier is a distinct entity from the first and second participants.